

Remarks

The amendments to the claims

5 Applicants have amended independent claims 5, 9,10, 11, 22, 26, 30, and 31 to better conform to what is disclosed in the patent application as filed. The amendment makes it clear that it is the messages of the *transaction* that are augmented. That this is so is particularly clearly set forth in the flowchart of FIG. 5 and in the discussion of FIG. 5 at page 19, lines 16-31. The claims as amended are thus fully supported by the Specification as filed. Applicants do not
10 believe that the amendments raise new issues. They do serve to put the claims in better condition for appeal, should that be necessary, and Applicants respectfully request that Examiner employ his discretionary powers under 37 C.F.R. 1.116 to enter the amendments.

Traversal of the rejections

15 *Explanation of the invention: FIGs. 4 and 5 and page 19, line 16-page 21, line 5*

A problem in any distributed system is making sure that changes in the components of the digital systems do not result in inconsistencies in the distributed system. The transactions that make the changes often employ protocols such as the two-phase commit protocol to ensure consistency. These protocols represent system overhead, i.e., they require the use of resources
20 in the distributed system that would otherwise be available to users of the distributed system. For that reason, various techniques have been developed for optimizing the protocols.

The invention is a novel technique for optimizing such protocols. The technique is based on the notion that the overhead required for a protocol can be reduced if the component of the system that is managing the protocol has available to it constantly-updated information that is relevant
25 to the protocol and uses that information to optimize the protocol.

In the exemplary embodiment, the distributed system is a distributed database system and the components are component database systems. The two-phase commit protocol is used to ensure that when a transaction involving copies of writable data in the component database systems is finished, all of the copies of the writable data are consistent. A characteristic of many
30 distributed database systems is that they include some components in which data involved in a

given transaction is read only and others in which it is writable. If the component database system that is managing the transaction knows which components are writable with regard to the transaction, the managing database system can optimize the two-phase commit protocol with regard to the read only components. Since the read only property of a component with regard to a component changes dynamically, what is needed is a technique which permits information as to the status of the component to be constantly returned to the managing database system.

As shown in FIG. 4, the technique has two components: augmented messages and retained state. The messages that are sent from component systems to the managing system during the transaction are *augmented* with information that indicates whether the component system sending the message is currently read-only with regard to the transaction. The *augmented message*, shown in FIG. 4 at 401, consists of a message that would be normally sent from the component system to the managing system during the transaction (403) to which cohort status information 405 has been added. Included in cohort status information 405 is a read only flag 407 which indicates whether the sending component system is currently read only with regard to the transaction.

The *retained state* is state for the transaction in the managing system which is updated from each augmented message for the transaction. In a preferred embodiment, the managing system maintains an outgoing link-transaction object 411 for each transaction for each of the component systems that are involved in a transaction. Included in a link 411(i) for a given transaction and component system is current cohort status 413(i), which embodies the retained state. Included in this state is a read only status flag 415(i), which is set from read-only status flag 407 each time a message for the link's transaction is received from the link's component system. The managing system uses the retained state to optimize the two-phase commit protocol.

Fig. 5 shows how this is done. At the beginning of the transaction, the managing system sets status flag 415(i) in the outgoing link transaction object to *read only* (505). Then, until the transaction is terminated (507), loop 513 is executed, in which the managing system receives each augmented message from the link transaction object's component system (515) and sets cohort status information 413 from the cohort status information 405 in the augmented message. On termination of the transaction (509), the managing system reads the current cohort status 413 for the transaction (519). If it is read only, the managing system optimizes the two-phase

commit protocol by only sending the protocol's *abort* message to the component system (527); otherwise, it does the entire two-phase commit protocol with the cohort (529).

The independent claims

The independent claims claim the optimization technique both from the point of view of the
5 managing system and from the point of view of the component system. Claim 5 as currently amended is an example of a claim written from the point of view of the component system:

5. (currently amended) A method of ensuring that a first component of a distributed system that exchanges messages that belong to a transaction with one or more other components of the distributed system is additionally aware of a state of an other component, the state being relevant to a protocol that is employed with the transaction and
10 the method comprising the steps practiced in the other component of:
determining the relevant state; and
augmenting a message of the transaction with state information indicating
15 the relevant state of the other component,
the first component using the state information to optimize the protocol.

The claimed method's purpose is to "ensur[e] that a first component of a distributed system that exchanges messages that belong to a transaction with one or more other components of the distributed system is additionally aware of a state of an other component . . . [that is] relevant to
20 a protocol . . .". The purpose is achieved by two steps practiced "in the other component": "determining the relevant state" and "augmenting a message of the transaction . . .". The first component is then able to "us[e] the state information to optimize the protocol."

Claim 11 is an example of a claim written from the point of view of the managing system:

11. (currently amended) A method of ensuring that a first component of a distributed system that exchanges messages belonging to a transaction with one or more other components of the distributed system is additionally aware of a state of an other component, the state being relevant to a protocol employed with the transaction and
25 the method comprising the steps practiced in the first component of:
receiving an augmented one of the messages, the augmented message having been augmented by the other component to additionally contain information indicating the relevant state of the other component;
30 retaining the relevant state from the augmented message; and
using the retained relevant state to optimize the protocol.
35 The preamble is like that of claim 5, except that the method steps are "practiced in the first component" and the method steps are "receiving an augmented one of the messages" ,

"retaining the relevant state from the augmented message", and "using the retained relevant state to optimize the protocol".

The "augmented message" and "retained relevant state" terminology used in the claims has its origins in the Specification as filed. See for example, page 22, lines 1-7:

5 With protocols for transactions, the optimization technique can be used in any situation where the protocol can be optimized if global or local coordinators for the transaction have current knowledge during the transaction of state in the cohorts that is relevant to the optimization. The only requirements for the optimization are that the messages sent from the cohorts in the course of the
10 transaction are augmented with information concerning the relevant state and that the relevant state is available to the coordinator while the coordinator is performing the transaction.

As already indicated above, the "augmented message" appears at 401 in FIG. 4 and the "retained relevant state" appears at 413.

15

The claims and the references

All claims have been rejected as anticipated by both Lampson and Cabrera. The following discussion will begin with Lampson.

The disclosure of Lampson

20 As pointed out in Applicants' response to Examiner's first Office action, there is no disclosure anywhere in Lampson of a non-managing component of the distributed system *augmenting* messages used in a transaction to provide the managing component with state information relevant to the two-phase commit protocol, and as might be expected from that, no use in a managing component of state received in the augmented messages to optimize the protocol.
25 Instead, Lampson's managing component receives the information necessary for the two-phase protocol in "yes", "no" and "acknowledge" messages sent by the components.

Lampson discloses a number of optimizations of the two-phase commit protocol. Two that are of interest in the immediate circumstances are the use of a list of most recent transaction identifiers (TIDs) to determine whether a log write is necessary before the commit is done and
30 using a "read only" message to reduce the number of messages necessary in the protocol. The

first optimization is described beginning at col. 15, line 9. In the optimization, if there are too many transaction identifiers for uncommitted transactions, the managing component does a forced write to the log, to ensure that the state can be recovered in case of a crash. There is no description whatever of where the managing component gets the TID information, but it is clearly part of the transaction information, and thus is not information with which transaction messages have been augmented, as required by Applicants' claims. The second optimization is that when a non-managing component is read-only with regard to the transaction, it can send a "read-only" message to the managing component instead of a "yes" or "no" message, and the managing component can use its knowledge of the fact that the non-managing component is read only to optimize the two-phase commit protocol. The information that the non-managing component is read-only is not, however, sent using an augmented transaction message, as required by Applicants' claims.

In applying Lampson, Examiner cites col. 6, which describes how the two-phase commit protocol works. The cited location describes how the coordinator sends "prepare" messages to the other participants, how they send "yes" or "no" votes, how the coordinator sends a "commit" message to all participants if all the other participants vote "yes" and otherwise an "abort" message, and how the participants respond to the "commit" message by committing the transaction and sending an "acknowledge" message. None of the information sent from the participants to the coordinator is sent by augmenting a message belonging to the transaction, and consequently, nothing in col. 6 discloses the limitations of Applicants' claims.

Cabrera

In their response to the previous Office action, Applicants pointed out that

Cabrera discloses a technique for keeping track of which copies of data stored in a set of mirrored disks are currently up to date. To keep track, Cabrera associates a data structure of $2n$ bits with the data, where n is the number of mirrored disks. This data structure is termed a *cohort set*. Again, the technique does not employ state information received from augmented messages.

In his rebuttal, Examiner refers Applicants to col. 7, lines 5-32 of Cabrera. What is disclosed at that location is how Cabrera uses a two-phase update process to update cohort sets. Updating happens whenever a write to one of the set of mirrored disks fails. Presumably, the failure is indicated by an error message to controller 12. There is simply no disclosure at the cited location or anywhere else in Cabrera that the information used to set the cohort sets is provided

to the controller 12 as an augmented transaction message. There is further no disclosure that the information in the cohort sets is used to optimize a protocol. Since that is the case, Cabrera does not disclose the limitations of Applicants' independent claims and cannot serve as the foundation of a rejection of those claims under 35 U.S.C. 102.

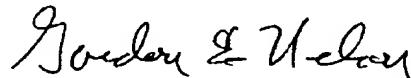
5 *The rejections of the dependent claims*

The dependent claims are all patentable because they are dependent from patentable claims. In addition, claims 2, 3, 6, 7, 23, 24, 27, and 28 all add details concerning the augmented messages and the use of the retained state from the augmented messages in optimizing the protocols, and are thus patentable in their own rights over references that show neither the augmented 10 messages nor the use of the retained state in optimizing the protocols. Claims 12-21 are dependent Beauregard claims and are patentable for the reasons that the claims they are dependent from are patentable.

Conclusion

Applicants have amended their claims to better conform to their Specification as filed and have 15 traversed Examiner's rejections of the claims. Applicants do not believe that the amendments raise any new issues. The amendments do serve to put the application in better condition for appeal, should that be necessary, and consequently, Applicants respectfully request that Examiner enter the amendment as provided in 37 C.F.R. 1.116 and allow the claims as amended. No fees are believed to be required for the amendment. Should any be, please 20 charge the additional fees to deposit account number 501315.

Respectfully submitted,



25

Attorney of record,
Gordon E. Nelson
57 Central St., P.O. Box 782
Rowley, MA, 01969,
Registration number 30,093
Voice: (978) 948-7632
Fax: (866) 723-0359

30

4/5/05
Date